

1

2 **CLAIMS:**

3 What is claimed is:

1 1. A method for controlling code in a multi-developer  
2 software development environment, the method comprising:  
3 identifying a plurality of software components as  
4 non-modifiable and preventing access and modification to  
5 the non-modifiable objects;  
6 receiving a request from a requesting user to modify  
7 one of the software components;  
8 determining whether the software component has been  
9 checked out by another user;  
10 providing the requesting user with a modifiable copy  
11 of the one of the software component if the software  
12 component has not been checked out by another user.

1 2. The method as recited in claim 1, further  
2 comprising:  
3 presenting the requesting user with an indication  
4 that the software component has been checked out by  
5 another user and is not available for modification if the  
6 object has been checked out by another user.

1 3. The method as recited in claim 1, further  
2 comprising:

3       creating a backup copy of the software component  
4 prior to providing the user with a modifiable copy of the  
5 software component.

1    4.    The method as recited in claim 1, further  
2 comprising:

3       updating a file indicating that the software  
4 component is checked out, the identity of the user that  
5 checked out the software component, the date the software  
6 component was checked out, and the time the software  
7 component was checked out.

1    5.    The method as recited in claim 1, further  
2 comprising:

3       prompting the requesting user to provide information  
4 about what is intended to be changed in the software  
5 component.

1    6.    The method as recited in claim 5, further  
2 comprising:

3       updating a file indicating the changes intended to  
4 be made to the software component.

1    7.    The method as recited in claim 1, wherein the  
2 software component is an object.

1    8.    The method as recited in claim 4, wherein the file  
2 is one of a table and a database.

1 9. The method as recited in claim 6, wherein the file  
2 is one of a table and a database.

1 10. The method as recited in claim 2, further  
2 comprising:

3 determining whether the requesting user wishes to  
4 send a message to the user who currently has the object  
5 checked out;

6 prompting the requesting user to indicate the  
7 message to be sent to the user who currently has the  
8 software component checked out; and

9 sending the message to the user who currently has  
10 the software component checked out.

1 11. The method as recited in claim 10, further  
2 comprising:

3 determining the method the requesting user wished  
4 the message to be sent to the user who currently has the  
5 software component checked out; and

6 sending the message by the method specified by the  
7 requesting user.

1 12. The method as recited in claim 1, further  
2 comprising:

3 determining that the requesting user requests to  
4 check the software component back in;

5 prompting the requesting user to enter a description  
6 of what has changed to the software component; and

7        saving an updated software component, indicating  
8        that the software component is checked in, and indicating  
9        that the software component is not modifiable.

1        13. The method as recited in claim 12, further  
2        comprising:

3               updating a file indicating that the software  
4        component is checked back in, the date and time checked  
5        in, the identity of the developer who checked the  
6        software component back in, and the nature of the changes  
7        made to the software component.

1        14. A computer program product in a computer readable  
2        media for use in a data processing system for controlling  
3        code in a multi-developer software development  
4        environment, the computer program product comprising:

5               first instructions for identifying a plurality of  
6        software components as non-modifiable and preventing  
7        access and modification to the non-modifiable objects;

8               second instructions for receiving a request from a  
9        requesting user to modify one of the software components;

10               third instructions for determining whether the  
11        software component has been checked out by another user;

12               fourth instructions for providing the requesting  
13        user with a modifiable copy of the one of the software  
14        component if the software component has not been checked  
15        out by another user.

1 15. The computer program product as recited in claim 14,  
2 further comprising:

3 fifth instructions for presenting the requesting  
4 user with an indication that the software component has  
5 been checked out by another user and is not available for  
6 modification if the object has been checked out by  
7 another user.

1 16. The computer program product as recited in claim 14,  
2 further comprising:

3 fifth instructions for creating a backup copy of the  
4 software component prior to providing the user with a  
5 modifiable copy of the software component.

1 17. The computer program product as recited in claim 14,  
2 further comprising:

3 fifth instructions for updating a file indicating  
4 that the software component is checked out, the identity  
5 of the user that checked out the software component, the  
6 date the software component was checked out, and the time  
7 the software component was checked out.

1 18. The computer program product as recited in claim 14,  
2 further comprising:

3 fifth instructions for prompting the requesting user  
4 to provide information about what is intended to be  
5 changed in the software component.

1 19. The computer program product as recited in claim 18,  
2 further comprising:

3 sixth instructions for updating a file indicating  
4 the changes intended to be made to the software  
5 component.

1 20. The computer program product as recited in claim 14,  
2 wherein the software component is an object.

1 21. The computer program product as recited in claim 17,  
2 wherein the file is one of a table and a database.

1 22. The computer program product as recited in claim 19,  
2 wherein the file is one of a table and a database.

1 23. The computer program product as recited in claim 15,  
2 further comprising:

3 sixth instructions for determining whether the  
4 requesting user wishes to send a message to the user who  
5 currently has the object checked out;

6 seventh instructions for prompting the requesting  
7 user to indicate the message to be sent to the user who  
8 currently has the software component checked out; and

9 eighth instructions for sending the message to the  
10 user who currently has the software component checked  
11 out.

1 24. The computer program product as recited in claim 23,  
2 further comprising:

3       ninth instructions for determining the computer  
4 program product the requesting user wished the message to  
5 be sent to the user who currently has the software  
6 component checked out; and  
7       tenth instructions for sending the message by the  
8 computer program product specified by the requesting  
9 user.

1   25. The computer program product as recited in claim 14,  
2 further comprising:

3       fifth instructions for determining that the  
4 requesting user requests to check the software component  
5 back in;

6       sixth instructions for prompting the requesting user  
7 to enter a description of what has changed to the  
8 software component; and

9       seventh instructions for saving an updated software  
10 component, indicating that the software component is  
11 checked in, and indicating that the software component is  
12 not modifiable.

1   26. The computer program product as recited in claim 25,  
2 further comprising:

3       eighth instructions for updating a file indicating  
4 that the software component is checked back in, the date  
5 and time checked in, the identity of the developer who  
6 checked the software component back in, and the nature of  
7 the changes made to the software component.

1 27. A system for controlling code in a multi-developer  
2 software development environment, the system comprising:  
3 first means for identifying a plurality of software  
4 components as non-modifiable and preventing access and  
5 modification to the non-modifiable objects;  
6 second means for receiving a request from a  
7 requesting user to modify one of the software components;  
8 third means for determining whether the software  
9 component has been checked out by another user;  
10 fourth means for providing the requesting user with  
11 a modifiable copy of the one of the software component if  
12 the software component has not been checked out by  
13 another user.

1 28. The system as recited in claim 27, further  
2 comprising:  
3 fifth means for presenting the requesting user with  
4 an indication that the software component has been  
5 checked out by another user and is not available for  
6 modification if the object has been checked out by  
7 another user.

1 29. The system as recited in claim 27, further  
2 comprising:  
3 fifth means for creating a backup copy of the  
4 software component prior to providing the user with a  
5 modifiable copy of the software component.



1 30. The system as recited in claim 27, further  
2 comprising:

3 fifth means for updating a file indicating that the  
4 software component is checked out, the identity of the  
5 user that checked out the software component, the date  
6 the software component was checked out, and the time the  
7 software component was checked out.

1 31. The system as recited in claim 27, further  
2 comprising:

3 fifth means for prompting the requesting user to  
4 provide information about what is intended to be changed  
5 in the software component.

1 32. The system as recited in claim 31, further  
2 comprising:

3 sixth means for updating a file indicating the  
4 changes intended to be made to the software component.

1 33. The system as recited in claim 27, wherein the  
2 software component is an object.

1 34. The system as recited in claim 30, wherein the file  
2 is one of a table and a database.

1 35. The system as recited in claim 32, wherein the file  
2 is one of a table and a database.

1 36. The system as recited in claim 28, further  
2 comprising:  
3 sixth means for determining whether the requesting  
4 user wishes to send a message to the user who currently  
5 has the object checked out;  
6 seventh means for prompting the requesting user to  
7 indicate the message to be sent to the user who currently  
8 has the software component checked out; and  
9 eighth means for sending the message to the user who  
10 currently has the software component checked out.

1 37. The system as recited in claim 36, further  
2 comprising:  
3 ninth means for determining the system the  
4 requesting user wished the message to be sent to the user  
5 who currently has the software component checked out; and  
6 tenth means for sending the message by the system  
7 specified by the requesting user.

1 38. The system as recited in claim 27, further  
2 comprising:  
3 fifth means for determining that the requesting user  
4 requests to check the software component back in;  
5 sixth means for prompting the requesting user to  
6 enter a description of what has changed to the software  
7 component; and  
8 seventh means for saving an updated software  
9 component, indicating that the software component is

10 checked in, and indicating that the software component is  
11 not modifiable.

1 39. The system as recited in claim 38, further  
2 comprising:

3 eighth means for updating a file indicating that the  
4 software component is checked back in, the date and time  
5 checked in, the identity of the developer who checked the  
6 software component back in, and the nature of the changes  
7 made to the software component.